UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 10/553,607 | 11/03/2006 | Wolfgang Becker | 09432.0062-00 | 6013 |

60668 7590 01/22/2009

SAP / FINNEGAN, HENDERSON LLP
901 NEW YORK AVENUE, NW
WASHINGTON, DC 20001-4413

| EXAMINER | |
|---|---|
| BROPHY, MATTHEW J | |

| ART UNIT | PAPER NUMBER |
|---|---|
| 2191 | |

| MAIL DATE | DELIVERY MODE |
|---|---|
| 01/22/2009 | PAPER |

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

| Office Action Summary | Application No. | Applicant(s) | |
|---|---|---|---|
| | 10/553,607 | BECKER ET AL. | |
| | Examiner | Art Unit | |
| | MATTHEW J. BROPHY | 2191 | |

*-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --*

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

1)☒ Responsive to communication(s) filed on *12 December 2008*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

4)☒ Claim(s) *1-25,28 and 29* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-25,28 and 29* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

**Application Papers**

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

## DETAILED ACTION

1. This office action is in response to amendment filed December 12, 2008

2. Claims 1, 21, 28 and 29 have been amended.

3. Amended Claim 23, Not previously addressed has been addressed below.

4.

### *Response to Amendment*

### *Claim Rejections - 35 USC § 112*

5. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

6. Claims 1-20 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Claim 1 recites the limitation "cyclically repeating the shifting and re-installing" in the newly added claim limitation. There is insufficient antecedent basis for this limitation in the claim.

### *Claim Rejections - 35 USC § 101*

7. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 28 and 29 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. The “computer-readable medium”

of these claims is interpreted to be directed to an electronic signal in view of applicant's specification ¶ 118 reproduced here:

> Carrier 970 is illustrated outside computer 900. For communicating CPP 100 to computer 900, carrier 970 is conveniently inserted into input device 940. Carrier 970 is implemented as any computer readable medium, such as a medium largely explained above (cf. memory 920). **Generally, carrier 970 is an article of manufacture having a computer readable medium with computer readable program code to cause the computer to perform methods of the present invention. Further, signal 980 can also embody computer program product 100.**
> (Applicant's Spec. ¶ 118 Oct. 18, 2005.)

An electronic signal is considered natural phenomenon, which is held to be a judicial exception category. Please see MPEP §2106.

## *Claim Rejections - 35 USC § 102*

8. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for a patent.

9.

***Claim Rejections - 35 USC § 103***

10. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

11. Claims 29 is rejected under 35 U.S.C. 103(a) as being unpatentable over PCT Patent Application Publication WO 03/005192 Reimer et al, hereinafter Reimer in view of US Patent 6,453,426 Gamache et al hereinafter Gamache

Regarding Claim 29, Reimer teaches: A computer-readable medium containing instructions for execution by a processor for the practice of a method for managing a computer system, the instructions being capable of causing a processor to: assign a service to a group of computers (**WO Page 4, Lines 23-27, "In this embodiment the method further comprises the steps of selecting the version of said at least one application adapted to the selected precon- figured operating system to be downloaded to the first server, downloading and installing the at least one application to the first memory location of the first server Further, the code means of the computer program product is arranged to make the computer perform the following steps: selecting the version of said at least one application adapted to the selected precon- figured operating system to be downloaded to the first server, downloading and installing the at least one application to the first memory location of the first server."**); shift a service that runs on a first computer of the group to run on a second computer in the group (**WO Page 4, Lines 23-27, "In this**

**embodiment the method further comprises the steps of selecting the version of said at least one application adapted to the selected precon- figured operating system to be downloaded to the first server, downloading and installing the at least one application to the first memory location of the first server Further, the code means of the computer program product is arranged to make the computer perform the following steps: selecting the version of said at least one application adapted to the selected precon- figured operating system to be downloaded to the first server, downloading and installing the at least one application to the first memory location of the first server.")**; and re-install the operating system to the first computer **(WO Page 15, Lines 14-15, "A server should be rebooted with another preconfigured operating system if, for instance, there is a need for more capacity in another preconfigured operating system.")**. cyclically repeating the shifting and re-installing for all computers in the group thereby **(Pg 18, Ln 18-21, " The invention also has the advantage that the life span of a server is increased due to the cycling of the servers (rebooting due to long up-time) and it also increases the security of the system since an uncorrupted version of the preconfigured operating system is downloaded each time the server is rebooted.")**

Reimer does not teach: keeping the number of computers that are re-installing the operating system smaller than the number of computers that are not re-installing the operating system. However, this limitation is taught by Gamache. **(Col. 7, Ln 53, Col 8, Ln 3, "In general, according to the GLUP protocol, one node (e.g. 58.sub.1 of FIG. 4A) serves as a "locker" node. The locker node 58.sub.1 ensures**

**that only one global update is in progress at any given time. With GLUP, a node (e.g., 58.sub.2) wishing to send an update to other nodes first sends a request to the locker node 58.sub.1. When any preceding updates are complete, the locker node 58.sub.1 gives permission for this "sender" node 58.sub.2 to broadcast its update to the other nodes in the cluster. In accordance with GLUP, the sender node 58.sub.2 sends the updates, one at a time, to the other nodes in a predetermined GLUP order that is ordinarily based on a unique number assigned to each node. GLUP can be utilized to replicate data to the machines of a cluster, including at least some of the cluster operational data, as described below. A more detailed discussion of the GLUP protocol is described in the publication entitled "Tandem Systems Review" Volume 1, Number 2, June, 1985 pp. 74-84, which is incorporated by reference herein.")** In addition, it would have been obvious to one of ordinary skill in the art at the time of the invention cycling of Reimer with the 'one at a time' updating protocol of Gamache as Gamache recognized (Col. 2, Ln 30-34) "...the performance and size of a cluster is limited by the rate at which the operational data can be updated, in part because such updates are relatively slow, comprising careful transactional logging of changes."

12. Claims 21, 22 and 24 are rejected under 35 U.S.C. 103(a) as being unpatentable over PCT Patent Application Publication WO 03/005192 Reimer et al, hereinafter Reimer in view of US Patent 5,551,047 Mori et al hereinafter Mori and further in view of US Patent 6,453,426 Gamache et al hereinafter Gamache

The text of these rejections not found in this office action can be found in the previous office action.

Regarding Claim 21, Reimer teaches: Method for managing a computer system, the system operating with a plurality of computers in at least one group, the method comprising: assigning a service to a group set of computers-(**WO Page 4, Lines 23-27, "In this embodiment the method further comprises the steps of selecting the version of said at least one application adapted to the selected precon- figured operating system to be downloaded to the first server, downloading and installing the at least one application to the first memory location of the first server Further, the code means of the computer program product is arranged to make the computer perform the following steps: selecting the version of said at least one application adapted to the selected precon- figured operating system to be downloaded to the first server, downloading and installing the at least one application to the first memory location of the first server.")**.;shifting a service that runs on a first computer of the group to run on a second computer in the group (**WO Page 4, Lines 23-27, "In this embodiment the method further comprises the steps of selecting the version of said at least one application adapted to the selected precon- figured operating system to be downloaded to the first server, downloading and installing the at least one application to the first memory location of the first server Further, the code means of the computer program product is arranged to make the computer perform the following steps: selecting the version of said at least one application adapted to the selected precon-**

**figured operating system to be downloaded to the first server, downloading and installing the at least one application to the first memory location of the first server.")**; and re-installing the operating system to the first computer **(WO Page 15, Lines 14-15, "A server should be rebooted with another preconfigured operating system if, for in- stance, there is a need for more capacity in another preconfigured operating system.")**. cyclically repeating the shifting and re-installing for all computers in the group **(Pg 18, Ln 18-21, " The invention also has the advantage that the life span of a server is increased due to the cycling of the servers (rebooting due to long up-time) and it also increases the security of the system since an uncorrupted version of the preconfigured operating system is downloaded each time the server is rebooted.")**

However, Reimer does not teach: testing the service in parallel operation on the first computer and on the second computer, and disabling the operation of the service by the first computer only if the test is successful. However, this limitation is taught by Mori. **(e.g. Col. 1, Line 62-Col 2 Line 8, "In a fault-free situation, both nodes will pass the acceptance test with the results computed with their first used versions. In such a case, the primary node notifies the shadow of its success in the acceptance test. Thereafter, only the primary node sends its output to the successor computing stations. However, if the primary node fails its test while the shadow node passes its test, the shadow node will take over the role of the primary as soon as it receives notice that the primary node has failed. If the primary node is completely lost, i.e., crashes, such that it is unable to notify the**

**shadow node of the failure of its test, the shadow node will recognize the failure of the primary upon the expiration of a present time limit."**

**And. E.g. Col. 3, Line 25-40, "The present method has several major characteristics. Parallel and asynchronous execution of multiple versions of a program module is performed with processors which are connected by a network. This includes the simple case where the same version of an application is allocated to multiple processors. There is no need for direct interaction between the processors during the execution of the same or different versions of the program module. The present method also utilizes two types of acceptance tests. One type is referred to as a result acceptance test for validation of the execution results of a program module and the other is referred to as an input acceptance test for validation and redundancy-detection of input data. These two types of acceptance tests are allocated to each processor together with a program module/version.")**

In addition, it would have been obvious to one of ordinary skill in the art to apply the acceptance test in Mori to the invention of Riemer, as the redundant acceptance testing of Mori allows that: (Col 3. Ln 6-7) "The occurrence of faults causes little or no delay to the application's computation" **(Pg 18, Ln 18-21, " The invention also has the advantage that the life span of a server is increased due to the cycling of the servers (rebooting due to long up-time) and it also increases the security of the system since an uncorrupted version of the preconfigured operating system is downloaded each time the server is rebooted.")**

None of the previous references teach: keeping the number of computers that are re-installing the operating system smaller than the number of computers that are not re-installing the operating system. However, this limitation is taught by Gamache. **(Col. 7, Ln 53, Col 8, Ln 3, "In general, according to the GLUP protocol, one node (e.g. 58.sub.1 of FIG. 4A) serves as a "locker" node. The locker node 58.sub.1 ensures that only one global update is in progress at any given time. With GLUP, a node (e.g., 58.sub.2) wishing to send an update to other nodes first sends a request to the locker node 58.sub.1. When any preceding updates are complete, the locker node 58.sub.1 gives permission for this "sender" node 58.sub.2 to broadcast its update to the other nodes in the cluster. In accordance with GLUP, the sender node 58.sub.2 sends the updates, one at a time, to the other nodes in a predetermined GLUP order that is ordinarily based on a unique number assigned to each node. GLUP can be utilized to replicate data to the machines of a cluster, including at least some of the cluster operational data, as described below. A more detailed discussion of the GLUP protocol is described in the publication entitled "Tandem Systems Review" Volume 1, Number 2, June, 1985 pp. 74-84, which is incorporated by reference herein.")** In addition, it would have been obvious to one of ordinary skill in the art at the time of the invention cycling of Reimer with the 'one at a time' updating protocol of Gamache as Gamache recognized (Col. 2, Ln 30-34) "…the performance and size of a cluster is limited by the rate at which the operational data can be updated, in part because such updates are relatively slow, comprising careful transactional logging of changes."

Claims 23 is rejected under 35 U.S.C. 103(a) as being unpatentable over PCT Patent Application Publication WO 03/005192 Reimer et al, hereinafter Reimer in view of US Patent 5,551,047 Mori et al hereinafter Mori and further in view of US Patent 6,453,426 Gamache et al hereinafter Gamache as applied to claim 1 above and further in view of US PG Pub 20020133537 Lau et al hereinafter Lau.

None of the previous references teach: copy a service that is running on the earlier detected blade from the earlier detected blade to the new blade. However, this limitation is taught by Lau. **(¶12 "There are a number of advantages associated with the inventions described herein. For example, when a server receives a client request for a data object that is not cached locally, the server may be able to obtain a copy of that data object from one of the other servers in the cluster instead of from the centralized storage device.")**

In addition, it would have been obvious to one of ordinary skill in the art to combine the teachings of Lau with the update system of Reimer as Lau teaches a system which obtains data objects directly from another server, avoiding the need to use the management server. (¶12) "As a result, the latencies attendant to obtaining data objects from a centralized storage device will be eliminated each time a server can bypass the centralized storage device and obtain a copy of a requested data object from the cache of another server in the cluster."

Regarding Claim 22, Reimer further teaches: shifting and re-installing is repeated cyclically for all computers in the groups

Reimer does not teach: thereby keeping the number of computers that are re-installing the operating system smaller than the number of computers that are not re-installing the operating system. However, this limitation is taught by Gamache **(Col. 7, Ln 53, Col 8, Ln 3, "In general, according to the GLUP protocol, one node (e.g. 58.sub.1 of FIG. 4A) serves as a "locker" node. The locker node 58.sub.1 ensures that only one global update is in progress at any given time. With GLUP, a node (e.g., 58.sub.2) wishing to send an update to other nodes first sends a request to the locker node 58.sub.1. When any preceding updates are complete, the locker node 58.sub.1 gives permission for this "sender" node 58.sub.2 to broadcast its update to the other nodes in the cluster. In accordance with GLUP, the sender node 58.sub.2 sends the updates, one at a time, to the other nodes in a predetermined GLUP order that is ordinarily based on a unique number assigned to each node. GLUP can be utilized to replicate data to the machines of a cluster, including at least some of the cluster operational data, as described below. A more detailed discussion of the GLUP protocol is described in the publication entitled "Tandem Systems Review" Volume 1, Number 2, June, 1985 pp. 74-84, which is incorporated by reference herein.").**

Regarding Claim 24, Reimer teaches: wherein the assigning step [assigning] is performed for services of a first class to on a first group of computers and for services of a second class on a second group of computers **(WO Page 11, Lines ,5-16, •"Figure 4 shows a. fourth embodiment of the present invention, comprising a large number of servers (S l- Sn), n=6, n is larger than 1, all connected to the load balance unit**

**LBU, as described in connection with figure 3, and to the storage unit SU. The function of the control unit CU and the monitoring unit MU is as described above, but the storage unit SU in this embodiment comprises a third memory location stor- ing information regarding, different applications for each standard operating system, In this example there are two different types of application configurations for each operating system: API (OSI), AP1 (OS1), AP2 (O82), and AP2 (OS2). Examples of application configurations may be Word processing applications, database applications, Economy applications, etc. Alternatively, for service providers on the internet, the applications may be e-commerce .applications, games or other programs, or applications providing information, such as news services.").**

13. Claims 1-8, 10, 12-20 are rejected under 35 U.S.C. 103(a) as being unpatentable over US PG Publication 2003/0140267 Abbondanzio et al hereinafter Abbondanzio in view of PCT Application Publication WO 03/005192 Reimer et al. hereinafter Reimer and further in view of hereinafter US Patent 5,551,047 Mori et al hereinafter Mori and further in view of US Patent 6,453,426 Gamache et al hereinafter Gamache

The text of these rejections not found in this office action can be found in a previous office action.

Regarding Claims 1 Abbondanzio teaches: A method for managing a computer system, the computer system operating with a plurality of blades, the method comprising: detecting the presence of a new blade in the computer system **(Abbondanzio Paragraph [0007] The problems identified above are in large part**

**addressed by a data processing system, referred to as a blade, comprising at least one main processor connected to a system bus, a system memory connected to the system bus and accessible to each of the main processors, a tamper mechanism, and a local service processor. The tamper mechanism is configured to change state each time the system is inserted into a slot in a rack enclosure. The local service processor on each blade is connected to the tamper mechanism and configured to update an insertion log upon detecting a change in state of the tamper mechanism. The insertion log provides a history of at least some rack insertions to which the system has been subjected. The system may include a non-volatile storage element accessible exclusively to the local service processor that contains the insertion log. The insertion log may include an insertion counter. In this embodiment, the local service processor is configured to increment the insertion counter upon each insertion. The local service processor may be further configured to issue an alert if the insertion counter exceeds a predetermined value. In one embodiment, the system further includes a battery backed real-time clock connected to the local service processor. The local service processor is configured to include real-time information corresponding to each insertion event in the insertion log. Each entry in the insertion log may include the identity of the rack enclosure and the geographical address of the slot of the corresponding insertion event. The local service processor may be configured to detect the tamper mechanism state and update the insertion following a power event such that the insertion log update is**

**independent of configuring the data processing system with a boot image.")**; Abbandanzio does not explicitly teach: automatically installing an operating system on the new blade; automatically configuring the operating system based on a configuration used in an earlier detected blade; and copying a service that is running on the earlier detected blade to the new blade.

However, these limitations is taught by Reimer:

automatically installing an operating system on the new blade **(WO Page 15, Lines 14-15, "A server should be rebooted with another preconfigured operating system if, for in- stance, there is a need for more capacity in another preconfigured operating system.")**; automatically configuring the operating system based on a configuration used in an earlier detected blade **(WO Page 13 Lines 21-23, "Step 62: The control unit CU uses the information obtained in step 61 to determine if a server needs to be rebooted and which preconfigured operating system should be used")**; and copying a service that is running on an earlier detected blade to the new blade (**WO Page 4, Lines 23-27, "In this embodiment the method further comprises the steps of selecting the version of said at least one application adapted to the selected precon- figured operating system to be downloaded to the first server, downloading and installing the at least one application to the first memory location of the first server Further, the code means of the computer program product is arranged to make the computer perform the following steps: selecting the version of said at least one application adapted to the selected precon- figured operating system to be downloaded to the first server, downloading and**

**installing the at least one application to the first memory location of the first server.")**. cyclically repeating the shifting and re-installing for all computers in the group **(Pg 18, Ln 18-21, " The invention also has the advantage that the life span of a server is increased due to the cycling of the servers (rebooting due to long up-time) and it also increases the security of the system since an uncorrupted version of the preconfigured operating system is downloaded each time the server is rebooted.")** In addition, it would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Abbondanzio with the OS updating techniques of Reimer, as Abbondanzio generally discusses the installation of Operating system and the update of software into new blades, while Reimer more specifically teaches a method of updating operating systems that allows the system to meet server demand.

None of the previous references teach: testing the service in parallel operation on the earlier detected blade and the new blade. However, this limitation is taught by Mori. **(e.g. Col. 1, Line 62-Col 2 Line 8, "In a fault-free situation, both nodes will pass the acceptance test with the results computed with their first used versions. In such a case, the primary node notifies the shadow of its success in the acceptance test. Thereafter, only the primary node sends its output to the successor computing stations. However, if the primary node fails its test while the shadow node passes its test, the shadow node will take over the role of the primary as soon as it receives notice that the primary node has failed. If the primary node is completely lost, i.e., crashes, such that it is unable to notify**

**the shadow node of the failure of its test, the shadow node will recognize the failure of the primary upon the expiration of a present time limit."**

**And. E.g. Col. 3, Line 25-40, "The present method has several major characteristics. Parallel and asynchronous execution of multiple versions of a program module is performed with processors which are connected by a network. This includes the simple case where the same version of an application is allocated to multiple processors. There is no need for direct interaction between the processors during the execution of the same or different versions of the program module. The present method also utilizes two types of acceptance tests. One type is referred to as a result acceptance test for validation of the execution results of a program module and the other is referred to as an input acceptance test for validation and redundancy-detection of input data. These two types of acceptance tests are allocated to each processor together with a program module/version.")**

In addition, it would have been obvious to one of ordinary skill in the art to apply the acceptance test in Mori to the invention of Riemer, as the redundant acceptance testing of Mori allows that: (Col 3. Ln 6-7) "The occurrence of faults causes little or no delay to the application's computation"

None of the previous references teach: keeping the number of computers that are re-installing the operating system smaller than the number of computers that are not re-installing the operating system. However, this limitation is taught by Gamache. **(Col. 7, Ln 53, Col 8, Ln 3, "In general, according to the GLUP protocol, one node**

**(e.g. 58.sub.1 of FIG. 4A) serves as a "locker" node. The locker node 58.sub.1 ensures that only one global update is in progress at any given time. With GLUP, a node (e.g., 58.sub.2) wishing to send an update to other nodes first sends a request to the locker node 58.sub.1. When any preceding updates are complete, the locker node 58.sub.1 gives permission for this "sender" node 58.sub.2 to broadcast its update to the other nodes in the cluster. In accordance with GLUP, the sender node 58.sub.2 sends the updates, one at a time, to the other nodes in a predetermined GLUP order that is ordinarily based on a unique number assigned to each node. GLUP can be utilized to replicate data to the machines of a cluster, including at least some of the cluster operational data, as described below. A more detailed discussion of the GLUP protocol is described in the publication entitled "Tandem Systems Review" Volume 1, Number 2, June, 1985 pp. 74-84, which is incorporated by reference herein.")** In addition, it would have been obvious to one of ordinary skill in the art at the time of the invention cycling of Reimer with the 'one at a time' updating protocol of Gamache as Gamache recognized (Col. 2, Ln 30-34) "…the performance and size of a cluster is limited by the rate at which the operational data can be updated, in part because such updates are relatively slow, comprising careful transactional logging of changes."

14. Claim 28 is rejected under 35 U.S.C. 103(a) as being unpatentable over US PG Publication 2003/0140267 Abbondanzio et al hereinafter Abbondanzio in view of PCT

Application Publication WO 03/005192 Reimer et al. hereinafter Reimer and further in view of hereinafter US Patent 5,551,047 Mori et al hereinafter Mori and further in view of US PG Pub 20020133537 Lau et al hereinafter Lau

Regarding Claims 28 Abbondanzio teaches: A method for managing a computer system, the computer system operating with a plurality of blades, the method comprising: detecting the presence of a new blade in the computer system **(Abbondanzio Paragraph [0007] The problems identified above are in large part addressed by a data processing system, referred to as a blade, comprising at least one main processor connected to a system bus, a system memory connected to the system bus and accessible to each of the main processors, a tamper mechanism, and a local service processor. The tamper mechanism is configured to change state each time the system is inserted into a slot in a rack enclosure. The local service processor on each blade is connected to the tamper mechanism and configured to update an insertion log upon detecting a change in state of the tamper mechanism. The insertion log provides a history of at least some rack insertions to which the system has been subjected. The system may include a non-volatile storage element accessible exclusively to the local service processor that contains the insertion log. The insertion log may include an insertion counter. In this embodiment, the local service processor is configured to increment the insertion counter upon each insertion. The local service processor may be further configured to issue an alert if the insertion counter exceeds a predetermined value. In one embodiment, the system further includes**

**a battery backed real-time clock connected to the local service processor. The local service processor is configured to include real-time information corresponding to each insertion event in the insertion log. Each entry in the insertion log may include the identity of the rack enclosure and the geographical address of the slot of the corresponding insertion event. The local service processor may be configured to detect the tamper mechanism state and update the insertion following a power event such that the insertion log update is independent of configuring the data processing system with a boot image.")**; Abbandanzio does not explicitly teach: automatically installing an operating system on the new blade; automatically configuring the operating system based on a confiquration used in an earlier detected blade; and copying a service that is running on the earlier detected blade to the new blade.

However, these limitations is taught by Reimer:

automatically installing an operating system on the new blade **(WO Page 15, Lines 14-15, "A server should be rebooted with another preconfigured operating system if, for in- stance, there is a need for more capacity in another preconfigured operating system.")**; automatically configuring the operating system based on a configuration used in an earlier detected blade **(WO Page 13 Lines 21-23, "Step 62: The control unit CU uses the information obtained in step 61 to determine if a server needs to be rebooted and which preconfigured operating system should be used") (WO Page 4, Lines 23-27, "In this embodiment the method further comprises the steps of selecting the version of said at least one application**

**adapted to the selected precon- figured operating system to be downloaded to the first server, downloading and installing the at least one application to the first memory location of the first server Further, the code means of the computer program product is arranged to make the computer perform the following steps: selecting the version of said at least one application adapted to the selected precon- figured operating system to be downloaded to the first server, downloading and installing the at least one application to the first memory location of the first server.")**. In addition, it would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Abbondanzio with the OS updating techniques of Reimer, as Abbondanzio generally discusses the installation of Operating system and the update of software into new blades, while Reimer more specifically teaches a method of updating operating systems that allows the system to meet server demand.

None of the previous references teach: testing the service in parallel operation on the earlier detected blade and the new blade. However, this limitation is taught by Mori. **(e.g. Col. 1, Line 62-Col 2 Line 8, "In a fault-free situation, both nodes will pass the acceptance test with the results computed with their first used versions. In such a case, the primary node notifies the shadow of its success in the acceptance test. Thereafter, only the primary node sends its output to the successor computing stations. However, if the primary node fails its test while the shadow node passes its test, the shadow node will take over the role of the primary as soon as it receives notice that the primary node has failed.**

**If the primary node is completely lost, i.e., crashes, such that it is unable to notify the shadow node of the failure of its test, the shadow node will recognize the failure of the primary upon the expiration of a present time limit."**

**And. E.g. Col. 3, Line 25-40, "The present method has several major characteristics. Parallel and asynchronous execution of multiple versions of a program module is performed with processors which are connected by a network. This includes the simple case where the same version of an application is allocated to multiple processors. There is no need for direct interaction between the processors during the execution of the same or different versions of the program module. The present method also utilizes two types of acceptance tests. One type is referred to as a result acceptance test for validation of the execution results of a program module and the other is referred to as an input acceptance test for validation and redundancy-detection of input data. These two types of acceptance tests are allocated to each processor together with a program module/version.")**

In addition, it would have been obvious to one of ordinary skill in the art to apply the acceptance test in Mori to the invention of Riemer, as the redundant acceptance testing of Mori allows that: (Col 3. Ln 6-7) "The occurrence of faults causes little or no delay to the application's computation"

None of the previous references teach: copy a service that is running on the earlier detected blade from the earlier detected blade to the new blade. However, this limitation is taught by Lau. **(¶12 "There are a number of advantages associated with**

**the inventions described herein. For example, when a server receives a client request for a data object that is not cached locally, the server may be able to obtain a copy of that data object from one of the other servers in the cluster instead of from the centralized storage device.")**

In addition, it would have been obvious to one of ordinary skill in the art to combine the teachings of Lau with the update system of Reimer as Lau teaches a system which obtains data objects directly from another server, avoiding the need to use the management server. (¶12) "As a result, the latencies attendant to obtaining data objects from a centralized storage device will be eliminated each time a server can bypass the centralized storage device and obtain a copy of a requested data object from the cache of another server in the cluster."

15. Claims 25 is rejected under 35 U.S.C. 103(a) as being unpatentable over PCT Application Publication WO 03/005192 Reimer et al. hereinafter Reimer as applied to claim 21 above in view of US PG Publication 2003/0140267 Abbondanzio et al. and US Patent 6,453,426 Gamache et al hereinafter Gamache further in view of hereinafter US Patent 5,551,047 Mori et al hereinafter Mori.

The text of these rejections not found in this office action can be found in the previous office action.

16. Claim 9 rejected under 35 U.S.C. 103(a) as being unpatentable over Abbondanzio et al hereinafter Abbondanzio in view of PCT Application Publication WO 03/005192 Reimer et al. hereinafter Reimer and further in view of hereinafter US Patent

5,551,047 Mori et al hereinafter Mori US Patent 6,453,426 Gamache et al hereinafter Gamache as applied to claim 1,6 above, and further in view of US PG Publication 2004/0255191 Fox et al hereinafter Fox.

The text of these rejections not found in this office action can be found in the previous office action.

17.

Claim 11 rejected under 35 U.S.C. 103(a) as being unpatentable over Abbondanzio et al hereinafter Abbondanzio in view of PCT Application Publication WO 03/005192 Reimer et al. hereinafter Reimer and further in view of hereinafter US Patent 5,551,047 Mori et al hereinafter Mori and further in view of US Patent 6,453,426 Gamache et al hereinafter Gamache as applied to claim 1 above, and further in view of US PG Publication 2003/0046394 Goddard et al hereinafter Goddard.

18. The text of these rejections not found in this office action can be found in the previous office action.

***Response to Arguments***

19. Applicant's arguments, see Remarks, filed December 12, 2008, with respect to the rejection(s) of claim(s) 23 as being unaddressed have been fully considered and are persuasive. Therefore, the rejection has been withdrawn. However, upon further consideration, a new ground(s) of rejection is made.

## *Conclusion*

20.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to MATTHEW J. BROPHY whose telephone number is 571-270-1642. The examiner can normally be reached on Monday-Thursday 8:00AM-5:00 PM EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Wei Zhen can be reached on (571) 272-3708. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

MJB

12/12/2008

/Wei Y Zhen/
Supervisory Patent Examiner, Art Unit 2191